(19)日本国特許庁（ＪＰ）　(12) **公開特許公報**（Ａ）　(11)特許出願公開番号
特開2000－47833
（P2000－47833A）
(43)公開日　平成12年２月18日(2000.2.18)

(51)Int.Cl.7		識別記号	FI		テーマコード*(参考)
Ｇ０６Ｆ	3/06	５４０	Ｇ０６Ｆ	3/06	５４０
	11/10	３３０		11/10	３３０Ｑ
Ｇ１１Ｂ	20/18	５４２	Ｇ１１Ｂ	20/18	５４２Ｚ
		５７０			５７０Ｚ
Ｈ０３Ｍ	13/00		Ｈ０３Ｍ	13/00	

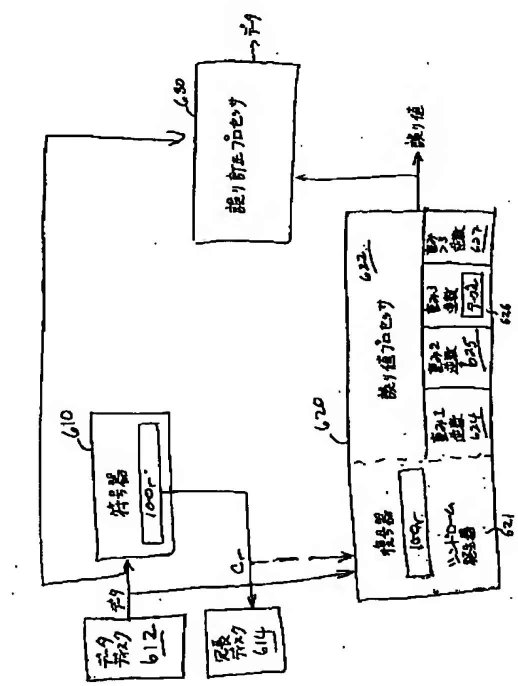
審査請求　未請求　請求項の数14　ＯＬ　（全 15 頁）

(21)出願番号　特願平11－135424
(22)出願日　平成11年５月17日(1999.5.17)
(31)優先権主張番号　０９／０７９７５７
(32)優先日　平成10年５月15日(1998.5.15)
(33)優先権主張国　米国（ＵＳ）

(71)出願人　591179352
クウォンタム・コーポレイション
QUANTUM CORPORATION
アメリカ合衆国、95035　カリフォルニア州、ミルピタス、マッカーシー・ブールバード、500
(72)発明者　リィ・ジィ・ウェン
アメリカ合衆国、01545　マサチューセッツ州、シュルーズベリー、ブルックデール・サークル、95
(74)代理人　100064746
弁理士　深見　久郎　（外３名）

(54)【発明の名称】　符号化のためのシステムおよび復号器

(57)【要約】
【課題】　ＲＡＩＤ－６または複数トラックテープシステムのための符号化／復号化システムを提供する。
【解決手段】　ＲＡＩＤ－６または複数トラックテープシステムのための符号化／複合化システムは、mについて選択された値の組のうち１つを用いる。m＋１は素数であり、体ＧＦ（2^m）は既約多項式$g(x)=x^m+x^{m-1}+...+x^2+x+1$によって生成される。このシステムは、ガロア体乗算を、循環シフトおよび排他的ＯＲ演算の組合せとして行ない、種々の（m＋１）ビットシンボルを２の選択された累乗に累乗することによって重み１、２および３の（m＋１）ビットシンボルの乗法逆元を決定する。このシステムを用いると、mの値を、セクタまたはテーププロックと同じまたはこれよりも大きなものとして選択することができる。



【特許請求の範囲】

【請求項１】 ｋのデータディスクからのｍビットデータシンボルを符号化してｒの冗長ディスクのためのｍビット冗長シンボルを生成することを、ｋのデータディスク各々からの対応するシンボルを符号化することにより行なう、符号化のためのシステムであって、前記システ*

*ムは、

（Ａ） ｋのデータディスクの対応するセクタからデータを受取るための手段と、

（Ｂ） （ｍ＋１）ビット冗長シンボルを、以下の式すなわち

【数１】

$$c_y = \alpha^{y*(k-1)} i_{k-1} + \alpha^{y*(k-2)} i_{k-2} + \ldots + \alpha^{y*(1)} i_1 + i_0$$

として生成するための符号化手段とを含み、式中ｙ＝０、１、…ｒ－１であり、i_jはｊ番目のデータディスクからのデータであり、αはｍ＋１を素数としたＧＦ（2^m）の要素の（ｍ＋１）ビット表現であり、前記手段は各c_yについて、

（ｉ） 乗算をｙビット循環シフトとして行なうためのシフトレジスタと、

（ｉｉ） シフトレジスタの内容に次のデータディスクからの対応するデータを加算するための加算器とを備え、前記システムはさらに、

（Ｃ） （ｍ＋１）ビット冗長シンボルをｍビット冗長シンボルに変換するための変換手段と、

（Ｄ） ｒの冗長ディスクの対応するセクタにｍビット冗長シンボルをストアするための手段とを含む、符号化のためのシステム。

【請求項２】 符号化手段はｍビットセクタを符号化す※

※る、請求項１に記載のシステム。

【請求項３】 シフトレジスタは、１つのシフトでｙビット循環シフトの結果を生成するフィードバック経路を含む（ｍ＋１）ビットシフトレジスタである、請求項１に記載のシステム。

【請求項４】 （ｋ＋ｒ）トラックテープのｋのトラックからのｍビットデータシンボルを符号化してｒの冗長トラックのためのｍビット冗長シンボルを生成することを、ｋのデータトラック各々からの対応するシンボルを符号化することにより行なう、符号化のためのシステムであって、

（Ａ） ｋのデータトラックの対応するブロックからデータを受取るための手段と、

（Ｂ） （ｍ＋１）ビット冗長シンボルを、以下の式すなわち

【数２】

$$c_y = \alpha^{y*(k-1)} i_{k-1} + \alpha^{y*(k-2)} i_{k-2} + \ldots + \alpha^{y*(1)} i_1 + i_0$$

として生成するための符号化手段とを含み、式中ｙ＝０、１、…ｒ－１であり、i_jはｊ番目のデータトラックからのデータであり、αはｍ＋１を素数としたＧＦ（2^m）の要素の（ｍ＋１）ビット表現であり、前記手段は各c_rについて、

（ｉ） 乗算をｙビット循環シフトとして行なうためのシフトレジスタと、

（ｉｉ） シフトレジスタの内容に次のデータトラックからの対応するデータを加算するための加算器とを備え、前記システムはさらに、

（Ｃ） （ｍ＋１）ビット冗長シンボルをｍビット冗長シンボルに変換するための変換手段と、

（Ｄ） ｍビット冗長シンボルをｒの冗長トラックの対応するブロックにストアするための手段とを含む、符号化のためのシステム。★

★【請求項５】 符号化手段はｍビットブロックを符号化する、請求項４に記載のシステム。

【請求項６】 シフトレジスタは、１つのシフトにおいてｙビット循環シフトの結果を生成するフィードバック経路を含む（ｍ＋１）ビットシフトレジスタである、請求項４に記載のシステム。

【請求項７】 ｒの冗長ディスクの対応するセクタにストアされた冗長シンボルを用いてｋのデータディスクの対応するセクタにおける誤りを訂正するための復号器であって、

（Ａ） ｋのデータディスクからデータを受取るための手段と、

（Ｂ） （ｍ＋１）ビット冗長シンボルを、以下の式すなわち

【数３】

$$c_y = \alpha^{y*(k-1)} i_{k-1} + \alpha^{y*(k-2)} i_{k-2} + \ldots + \alpha^{y*(1)} i_1 + i_0$$

として生成するための復号化手段とを含み、式中ｙ＝０、１、…ｒ－１であり、i_jはｊ番目のデータディスクからの誤りのないデータであり、またはｊ番目のディスクのデータに誤りがあればすべて０であり、αはｍ＋１を素数としたＧＦ（2^m）の要素の（ｍ＋１）ビット表現であり、前記手段は各c_rについて、

（ｉ） 乗算をｙビット循環シフトとして行なうためのシフトレジスタと、

（ｉｉ） シフトレジスタの内容に次のデータディスクからのデータを加算するための加算器とを備え、前記復号器はさらに、

（Ｃ） ｚを誤りのあるデータを含むディスクの数として、シンドロームS_j、S_{j+1}、…S_{j+z-1}を生成するためのシンドローム発生器を含み、前記シンドローム発生

器は、 ＊ ＊【数４】

$$S_j=(\alpha^j)^{e0}*i'_{c0}+(\alpha^j)^{c1}*i'_{e1}+\ldots+(\alpha^j)^{cz}*i'_{ez}$$

を生成し、式中ｊ＝０、１、…、ｒ－１であり、i'_{ez}はｚ番目の誤りのあるディスクと関連する誤りパターンであり、S_jはｊ番目の冗長ディスクと関連するシンドロームであり、前記復号器はさらに、

（Ｄ） シンドロームと関連する誤り値を決定するための手段を含み、前記手段は、重み１の（ｍ＋１）ビットシンボル$(\alpha i)^p$の乗法逆元をα^{m+1-ip}として決定するための手段を含み、前記復号器はさらに、

（Ｅ） （ｍ＋１）ビット誤り値をｍビット誤り値に変換するための変換手段と、

（Ｆ） ｍビット誤り値を用いて誤りのあるｍビットシンボルを訂正するための手段とを含む、復号器。

【請求項８】 誤り値を決定するための手段はさらに、$A=\alpha^h+\alpha^v$、重み２の（ｍ＋１）ビットシンボルの乗法逆元を、$A^{-1}=(\alpha^{m+1-h})*\delta^{v-h}$として決定するための手段を含み、式中$\delta$は$\alpha^{m-1}+\alpha^{m-3}+\cdots+\alpha^3+\alpha^1$である、請求項７に記載の復号器。

【請求項９】 誤り値を決定するための手段はまた、Ａ※※＝$\alpha^a+\alpha^b+\alpha^c$、重み３の（ｍ＋１）ビットシンボルの乗法逆元を$\alpha^{c}*D^{b-c}$として決定するための手段を含み、式中Ｄは$\alpha^{a'}+\alpha+1$の逆数であり、$a'=(a-c)/(b-c)$である、請求項８に記載の復号器。

【請求項１０】 誤り値を決定するための手段はさらに、…１…００１１の形式の重み３の（ｍ＋１）ビットシンボルの乗法逆元の（ｍ－１）要素テーブルを含み、前記手段はテーブルを投入してシンボルＤを決定する、請求項９に記載の復号器。

【請求項１１】 ｒの冗長トラックの対応するブロックにストアされた冗長シンボルを用いて複数トラックテープシステムのｋのデータトラックの対応するブロックにおける誤りを訂正するための復号器であって、

（Ａ） ｋのデータトラックからデータを受取るための手段と、

（Ｂ） （ｍ＋１）ビット冗長シンボルを、以下の式すなわち

【数５】

$$c_y=\alpha^{y*(k-1)}i_{k-1}+\alpha^{y*(k-2)}i_{k-2}+\ldots+\alpha^{y*(1)}i_1+i_0$$

として生成するための復号化手段とを含み、式中、ｙ＝０、１、…ｒ－１であり、i_jはｊ番目のデータトラックからの誤りのないデータであり、またはｊ番目のトラックのデータに誤りがあればすべて０であり、αはｍ＋１を素数とするＧＦ（2^m）の要素の（ｍ＋１）ビット表現であり、前記手段は各c_rについて、

（ｉ） 乗算をｒビット循環シフトとして行なうためのシフトレジスタと、★

★（ｉｉ） シフトレジスタの内容に次のデータトラックからのデータを加算するための加算器とを備え、前記復号器はさらに、

（Ｃ） シンドロームS_j、S_{j+1}、…S_{j+z-1}を生成するためのシンドローム発生器を含み、式中ｚは誤りのあるデータを含むトラックの数であり、シンドローム発生器は、

【数６】

$$S_j=(\alpha^j)^{e0}*i'_{c0}+(\alpha^j)^{el}*i'_{cl}+\ldots+(\alpha^j)^{ez}*i'_{ez}$$

を生成し、式中ｊ＝０、１、…、ｒ－１であり、i'_{ez}はｚ番目の誤りのあるトラックに関連する誤りパターンであり、S_jはｊ番目の冗長トラックに関連するシンドロームであり、前記復号器はさらに、

（Ｄ） シンドロームに関連する誤り値を決定するための手段を含み、前記手段は重み１の（ｍ＋１）ビットシンボル$(\alpha i)^p$の乗法逆元をα^{m+1-ip}として決定するための手段を含み、前記復号器はさらに、

（Ｅ） （ｍ＋１）ビットエラー値をｍビットエラー値に変換するための変換手段と、

（Ｆ） ｍビット誤り値を用いて誤りのあるｍビットシンボルを訂正するための手段とを含む、復号器。

【請求項１２】 誤り値を決定するための手段はさらに、$A=\alpha^h+\alpha^v$、重み２の（ｍ＋１）ビットシンボルの乗法逆元を、$A^{-1}=(\alpha^{m+1-h})*\delta^{v-h}$として決定するための手段を含み、式中$\delta$は$\alpha^{m-1}+\alpha^{m-3}+\cdots+\alpha^3+\alpha^1$である、請求項１１に記載の復号器。

【請求項１３】 誤り値を決定するための手段はまた、$A=\alpha^a+\alpha^b+\alpha^c$、重み３の（ｍ＋１）ビットシンボルの乗法逆元を$\alpha^{c}*D^{b-c}$として決定するための手段を含み、式中Ｄは$\alpha^{a'}+\alpha+1$の逆数であり、$a'=(a-c)/(b-c)$である、請求項１２に記載の復号器。

【請求項１４】 誤り値を決定するための手段はさらに、…１…００１１の形式の重み３の（ｍ＋１）ビットシンボルの乗法逆元の（ｍ－１）要素テーブルを含み、前記手段はテーブルを投入してシンボルＤを決定する、請求項９に記載の復号器。

【発明の詳細な説明】

【０００１】

【発明の背景】複数の独立したディスクを備え、冗長データをストアするためにこれらのディスクのうち２つ以上を使用するシステムは一般に、ＲＡＩＤ－６システムと呼ばれており、ＲＡＩＤは、「廉価または独立ディス

クの冗長アレイ（redundantarray ofinexpensive or independentdisks）」の頭字語である。「ｋ」のデータディスクおよび「ｒ」の冗長ディスクを備えるシステムにおいては、ｒの冗長ディスクに記録されたデータは、ｋのデータディスクに記録されたデータを操作することにより生成される。典型的には、ｒの冗長ディスクのデータは、ｋのデータディスクのデータの誤り訂正符号化の結果である。

【０００２】ｋのデータディスクのデータはデータ符号語の形式を取り、符号語各々がデータシンボルおよび関連する誤り訂正符号（ＥＣＣ）シンボルを含む。ＥＣＣシンボルは、ＥＣＣに従い従来の態様でデータシンボルを符号化することによって生成される。ｒの冗長ディスクのセクタは、冗長符号語を含む。この符号語におけるシンボルは、ｋのデータディスク各々からの対応するシンボルを符号化することによって生成される。この符号化は、ｋの対応するシンボルの「カラム」の符号化と考えることができる。本明細書において混乱を避けるために、ｋのシンボルのカラムを符号化するのに使用されるＥＣＣは、「カラムＥＣＣ」と呼ぶことにする。典型的には、カラムＥＣＣは、ｋのデータディスクに対するデータ符号語を生成するために使用されるＥＣＣと同じではない。

【０００３】ｒの冗長ディスクにストアされたシンボルは、含まれるＥＣＣシンボルを用いて訂正できないデータ符号語における誤りを訂正するために使用される。システムは、誤りを訂正するために、カラムＥＣＣに従い、誤りのあるデータシンボルを含むカラムを復号化する。

【０００４】システムは、冗長ディスクのためのデータを生成するために、ｋの対応するセクタにストアされたデータをｍビットシンボルに分割する。このシステムは次に、カラムＥＣＣに従いｋのセクタ各々からの対応するｍビットシンボルを符号化して、ｒの対応するｍビット冗長シンボル生成する。これらのｒのシンボルはそれぞれ、ｒの冗長ディスク各々の対応するセクタにおいて記録される。

【０００５】システムの設計者は、カラムの符号化および復号化に必要な時間を最短にするために、比較的大きなシンボルサイズ、すなわち「ｍ」についての値を選択しようとするものである。こうすると、１セクタ当りに必要なカラム符号化／復号化動作数が減少する。符号化および復号化動作には、複数のガロア体乗算が必要である。従来のシステムでは、ｍのサイズが増大するほど、ガロア体乗算の複雑度も大幅に増大する。さらに、各復号化動作は、ガロア体ＧＦ（2^m）の要素の乗法逆元の決定を含む。ｍが大きい場合、乗法逆元を発見する演算には時間がかかり、かつ／または、大きな、2^m-1要素の、ルックアップテーブルを必要とするかもしれない。したがって、ｍの値は、システムの複雑さと符号化

／復号化に要する全時間との歩み寄りにおいて選択される。

【０００６】複数トラックテープシステムのトラックを、ＲＡＩＤ－６システムにおいて複数のディスクを使用するのと同じやり方で使用してもよい。したがって、データ符号語はｋのトラックにストアされ、関連の冗長シンボルは残りのｒのトラックにストアされる。ｋ＋ｒのディスクに対して使用されるのと同じ符号化／復号化技術を、ｋ＋ｒのトラックに対して用いる。このようにして、カラム符号化に際しｍビットシンボルのサイズを選択する上で、同じ歩み寄りを行なう。

【０００７】

【発明の概要】本発明は、ｍについて選択された値の組のうち１つを使用し、ガロア体乗算を循環シフトおよび排他的ＯＲ演算の組合せとして行なう、ＲＡＩＤ－６または複数トラックテープシステムのための符号化／復号化システムである。このシステムはまた、以下でより詳細に述べるように、ある乗法逆演算を、本質的には体の種々の要素を２の選択された累乗に累乗することによって決定する。このシステムを用いると、ｍの値を、システムが実行する乗算または逆演算の複雑性を増すことなく、任意に大きなものとして選択することができる。

【０００８】このシステムは、（ｍ＋１）ビットシンボルを用い、乗算および逆演算を行なう。ｍ＋１は素数である。ｍの値は、体ＧＦ（2^m）を以下の既約多項式によって生じさせることができるように選択される。

【０００９】$g(x)=x^m+x^{m-1}+...+x^2+x+1$

ＧＦ（2^m）の２つの要素ａ（ｘ）およびｂ（ｘ）の乗算を、循環シフトされた要素ｂ（ｘ）の（ｍ＋１）ビット版の排他的ＯＲを取ることによって行なうことができる。さらに、以下で述べるように、重み１、２または３のＧＦ（2^m）の要素ｃ（ｘ）の乗法逆元を、（ｍ＋１）ビット表現を用いて、本質的には体の特定の要素を２の関連の累乗に累乗することによって容易に決定することができる。このように、既知の先行技術のシステムよりも、復号化動作の複雑性は低く、かつ／または要する時間も短い。

【００１０】このシステムを用いると、ｍの値を、たとえばディスクのセクタまたはテープのブロックの大きさになるように選択できる。このようにして、カラム誤り訂正符号化／復号化を、１セクタまたはブロック当り１回行なうことができる。

【００１１】本発明についての以下の説明では、添付の図面を参照する。

【００１２】

【実施例の詳細な説明】本明細書では、ｋのデータディスクおよびｒの冗長ディスクを備えた複数ディスクシステムと関連付けて、本発明について説明を行なう。本発明を、ｋのデータトラックおよびｒの冗長トラックを備える複数トラックテープシステムについて使用すること

もできる。

【００１３】従来のＲＡＩＤ－６システムでは、ｋのセクタの各々におけるｉ番目のｍビットシンボル、すなわちｉ番目のカラムと関連するｒの冗長シンボルは、典型的には以下のようにして生成される。

【００１４】

【数７】

$$c_{0i} = i_0 + i_1 + i_2 + \ldots + i_{k-1}$$
$$c_{1i} = (\alpha^{1*0}) * i_0 + (\alpha^{1*1}) * i_1 + (\alpha^{1*2}) * i_2 + \ldots + (\alpha^{1*(k-1)}) * i_{k-1}$$
$$c_{2i} = (\alpha^{2*0}) * i_0 + (\alpha^{2*1}) * i_1 + (\alpha^{2*2}) * i_2 + \ldots + (\alpha^{2*(k-1)}) * i_{k-1}$$
$$\ldots$$
$$c_{(r-1)i} = (\alpha^{(r-1)*0}) * i_0 + (\alpha^{(r-1)*1}) * i_1 + (\alpha^{(r-1)*2}) * i_2 + \ldots + (\alpha^{(r-1)*(k-1)}) * i_{k-1}$$

【００１５】式中、$C_{(r-1)i}$は、ｒ番目の冗長ディスクのｉ番目のカラムに対する冗長シンボルであり、i_jは、ｊ番目のディスクのｉ番目のカラムにストアされたシンボルであり、「＋」および「＊」は、ガロア体加算および乗算を表わし、「α」はＧＦ（2^m）の要素である。このように、各カラムに対し冗長シンボルを生成するために行なわなければならない、複数の、時間の掛かるガロア体乗算がある。

【００１６】本発明のシステムは、選択された数の体ＧＦ（2^m）のうち１つを用い、ガロア体乗算を簡素化している。選択されたｍに対し、ｍ＋１を素数とすると、ガロア体ＧＦ（2^m）を、既約多項式、

$g(x) = x^m + x^{m-1} + \ldots + x^2 + x + 1$

によって生じさせることができ、要素２は、体ＧＦ（ｍ＋１）の原始元である。このような体は、本明細書に引用により援用する、Ｊ．Ｋ．ウルフ（J.K. Wolf）による、Discrete Mathematics 106/107(1992) 497-502頁において発表された、「ｍのある値に対しＧＦ（2^m）において乗算するための効率的な回路（Efficient Circuits For Multiplying In GF(2^m) For Certain Values of m）」と題された文献において論じられている。以下で述べるように、このシステムはこういった体を利用し、ＧＦ（2^m）の原始元が既知であることを必要としない。このことは、より大きな体を用いるときには重要となる。なぜなら、原始元を決定することは困難であり得るからである。ｍ＜３２９８７の場合のｍの値は、図１から３の表に列記されている。

【００１７】同時係属中の、「ＧＦ（2^{w+i}）のシンボルの（ｗ＋ｉ＋１）ビット表現を用いた修正リードソロモン誤り訂正システム（MODIFIED REED-SOLOMON ERROR CORRECTION SYSTEM USING (w+i+1)-bit REPRESENTATIONS OF SYMBOLS OF GF(2^{w+i})」と題された米国特許出願第０８／７８６，８９４号において述べられているように、ＧＦ（2^m）の要素は、補数である２つの（ｍ＋１）ビットシンボルａ（ｘ）およびｂ（ｘ）によって表現することができる。すなわち、ａ（ｘ）＋ｂ（ｘ）＝０である。たとえば、ＧＦ（2^4）の要素を２つの５ビットシンボルのいずれかによって表現することができる。

【００１８】

【表１】

要素	４ビット表現	５ビット表現
0	0000	00000 または 11111
β^0	0001	00001 または 11110
β^1	0011	00011 または 11100
β^2	0101	00101 または 11010
β^3	1111	01111 または 10000
...	...	...
β^{14}	1010	01010 または 10101

【００１９】ｍビットシンボルの（ｍ＋１）ビット表現は、先行ゼロを追加するかまたは先行１を追加し、残りのｍビットを補足することによって生成される。明らかに、一方の（ｍ＋１）ビット表現は、他方の（ｍ＋１）ビット表現よりも重み、すなわち１の数が少ない。

【００２０】２つの（ｍ＋１）ビットシンボルの乗算は、一方のシンボルの循環シフトされたコピーと、他方のシンボルの係数またはビットとの排他的ＯＲを取ることによって行なうことができる。例としてＧＦ（2^4）の要素を用いると、２つの要素ａ（ｘ）およびｂ（ｘ）の乗算すなわちａ（ｘ）＊ｂ（ｘ）は、$a(x) = a_4 a_3 a_2 a_1 a_0$、$b(x) = b_4 b_3 b_2 b_1 b_0$の場合以下のようにして行なわれる。

【００２１】

【数８】

$$b_0*(a_4a_3a_2a_1a_0)$$
$$+b_1*(a_3a_2a_1a_0a_4)$$
$$+b_2*(a_2a_1a_0a_4a_3)$$
$$+b_3*(a_1a_0a_4a_3a_2)$$
$$+b_4*(a_0a_4a_3a_2a_1)$$

【００２２】ｂ（ｘ）の係数がゼロならば、この係数とａ（ｘ）を循環シフトしたものとの積はゼロであり、したがって計算の必要がない。このようにして、乗算を、ｂ（ｘ）についての２つの（ｍ＋１）ビット表現の重みの低い方を選択することによって簡単にすることができる。実際に、この乗算を、２つのシンボルの重みの低い方としてｂ（ｘ）を選択することによってさらに簡単にすることができる。例として、ｂ（ｘ）＝００１０１の場合、乗算は、ａ（ｘ）＊ｂ（ｘ）＝（$a_4a_3a_2a_1a_0$）＋（$a_2a_1a_0a_4a_3$）となる。最も簡単な乗算は、１という１つの係数とｍのゼロの値の係数とを有する（ｍ＋１）ビットシンボルｂ（ｘ）を含む。この積はしたがって、ａ（ｘ）の（ｍ＋１）ビット表現を循環シフトさせたものとなる。

【００２３】この乗算は、ｍのサイズが増大しても複雑にならない。したがって、ｍを少なくともたとえばセクタのビット数の大きさとして選択することができる。このことは、１セクタに付き１つのカラム符号化動作を行なうことを意味している。

【００２４】（１）　符号化上記のように、従来のＲＡＩＤ－６システムでは、ｋのセクタの各々におけるｉ番目のｍビットシンボルと関連する、すなわちｉ番目のカラムと関連するｒの冗長シンボルは、典型的には以下のようにして生成される。式中、ｙ＝０、１．．．、ｒ－１である。

【００２５】

【数９】

$$c_{yi}=(\alpha^{y*0})*i_0+(\alpha^{y*1})*i_1+(\alpha^{y*2})*i_2+\ldots+(\alpha^{y*(k-1)})*i_{k-1}$$

【００２６】現システムは、好ましくは、少なくともディスクセクタにおけるビット数の大きさのｍの値を用いる。したがって、１つの符号化動作により、冗長ディスクにおける対応するセクタについてセクタサイズのシンボルが生まれる。要素αは、ｍ－１の先行ゼロを有する００００００…１０として選択される。冗長ディスクの数がｍ＋１以下であれば、冗長シンボルに対し等式の組において必要なαの累乗は以下のようになる。乗算は循環シフトである。

【００２７】

【数１０】

$$\alpha^0=000\ldots01$$
$$\alpha^1=000\ldots10$$
$$\alpha^2=000\ldots100$$
$$\vdots$$
$$\alpha^m=100\ldots00$$

【００２８】次に図４を参照すると、c_rについての等式におけるｋ番目の積は、回路１００$_r$において、ｋ番目のディスクの対応するセクタからのデータの（ｍ＋１）ビット表現を（ｍ＋１）ビットシフトレジスタ１１０ｒに与え、シンボルを左に「ｊ」ビットだけ循環シフトすることによって計算される。ｊ＝ｒ－１，ｒ－２…１，０である。この積を、次に、加算器１２０ｒにおいて、（ｋ－１）番目のディスクの対応するシンボルからのデータに加算する。再びシフトレジスタはｊビットだけシフトされ、（ｋ－２）番目のディスクの対応するセクタからのデータがシフトレジスタの内容に加算され、以降同様の動作を行ない、c_rの（ｍ＋１）ビット表現を生成する。

【００２９】$c_0=i_0+(\alpha^{0*1})\ i_1+(\alpha^{0*2})\ i_2+\ldots+(\alpha^{0*(k-1)})\ i_{k-1}$を生成する回路１００$_0$については、図５においてより詳細に示している。ｋ番目のディスクのセクタからのデータの（ｍ＋１）ビット表現は、シフトレジスタ１１０$_0$に与えられ、シフトレジスタはｊ＝０ビットシフトされる。（ｋ－１）番目のディスクのセクタ情報の（ｍ＋１）ビット表現は、それぞれのビットを組合せる加算器サブ回路１２１$_0$－１２１$_m$を含む加算器１２０$_r$においてシフトレジスタの内容に加算され、シフトレジスタはしたがって、$(\alpha^{0*1})*i_{k-1}+i_{k-2}$を含む。シフトレジスタを再びｊ＝０ビットシフトし、（ｋ－２）番目のディスクの対応するセクタからのデータの（ｍ＋１）ビット表現を次に、シフトレジスタの内容に加算する。すなわちシフトレジスタは以下を含む。

【００３０】

【数１１】

$$\alpha^0((\alpha^{0*1})i_{k-1}+i_{k-2})+i_{k-3}$$
または
$$(\alpha^{0*2})i_{k-1}+(\alpha^0)i_{k-2}+i_{k-3}$$

【００３１】シフトレジスタは再びｊ＝０ビットシフトされ、（ｋ－３）番目のディスクの対応するセクタのデータの（ｍ＋１）ビット表現をシフトレジスタの内容に加算する。すなわちレジスタは以下を含む。

【００３２】

【数１２】

$$\alpha^0((\alpha^{0*2})i_{k-1}+(\alpha^{0*1})i_{k-2}+i_{k-3})+i_{k-4}$$

または

$$(\alpha^{0*3})i_{k-1}+(\alpha^{0*2})i_{k-2}+(\alpha^0)i_{k-3}+i_{k-4}$$

【００３３】同様の動作を、残りのデータディスクの対応するセクタからのデータがシフトレジスタに加算されるまで行なう。結果、レジスタは（ｍ＋１）ビット冗長シンボルc_0を含む。

【００３４】冗長シンボルc_0の（ｍ＋１）ビット表現を次に、変換器１３０においてｍビットシンボルに変換する。変換器１３０は、（ｍ＋１）ビット表現の先頭ビットと残りｍビットの各々との排他的ＯＲを取る。先頭ビットがゼロであれば、残りのｍビットは変化なしであり、この演算では先頭ビットを無視した場合と同じ結果が生まれる。先頭ビットが１であれば、排他的ＯＲ演算は、ｍビットを補足し、先頭ビットは無視される。

【００３５】セクタデータの（ｍ＋１）ビット表現は、先頭ゼロビットを追加したセクタの内容である。このようにして、セクタから読出したｍビットを直接、ｍビットを（ｍ＋１）ビット表現に変換するという別のステップを行なわずに、シフトレジスタの適切なロケーションに与えることができる。

【００３６】図６に示すように、c_1を生成するための回路100_1も同様にシフトレジスタ110_1および加算器120_1を含む。ｋ番目のデータディスクのセクタからのデータをシフトレジスタに与え、レジスタはｊ＝１ビットシフトされ、（ｋ－１）番目のディスクの対応するセクタからのデータがシフトレジスタの内容に加算される。（ｋ－１）番目のシンボルのビット０はｋ番目のシンボルのビットｍと組合され、（ｋ－１）番目のシンボルのビット１はｋ番目のシンボルのビット０と組合され、以降同様に行なって、

$\alpha^j i_{k-1}+i_{k-2}$

を生成する。次に、シフトレジスタ110_1の内容をｊ＝１ビットシフトし、（ｋ－２）番目のデータディスクの対応するセクタからのデータをシフトレジスタの内容に加算して以下を生成する。

【００３７】

【数１３】

$$\alpha^j(\alpha^j i_{k-1}+i_{k-2})+i_{k-3}$$

または

$$(\alpha^{j*2})i_{k-1}+(\alpha^{j*1})i_{k-2}+i_{k-3}$$

【００３８】以降同様に行なう。ｋのディスクすべてからのデータをシフトレジスタに加算した後、内容は変換器１３０に与えられる。変換器は（ｍ＋１）ビットシンボルをｍビットシンボルc_1に変換する。

【００３９】ｊ＞１に対するシフトレジスタ110_jは、１つのクロックサイクルにおいてｊビットシフトを行なうように設定されたフィードバック経路を有してもよい。こうして、単一のシフト演算により適切なビット操作を行なってシフトレジスタ内容をα^jで乗算する。このようにして、冗長シンボル$c_0 \cdots c_{r-1}$を、同数のクロックサイクルで並行して生成することができる。

【００４０】次に図７を参照して、c_2を生成するための回路は、１クロックサイクルで２ビット循環シフトの結果を生成するようにされたフィードバック経路を備えたシフトレジスタ110_2を含む。シフトレジスタ110_2は、ビットｍをロケーション112_mからロケーション112_1に与え、ロケーション112_{m-1}のビットｍ－１をロケーション112_0に与え、残りのビットｍ－２、…０それぞれをロケーション112_{m-2}、…112_0からロケーション112_m、…、112_2に与える。（ｋ－１）番目のディスクの対応するセクタからのビットを次にシフトレジスタの内容に加算し、（ｋ－１）番目のセクタのビット０がｋ番目のセクタのビットｍ－１と組合され、（ｋ－１）番目のセクタのビット１がｋ番目のセクタのビットｍと組合され、（ｋ－１）番目のセクタのビット２がｋ番目のセクタのビット０と組合され、以降同様に行なわれるようにする。したがって、シフトレジスタは

$\alpha^j i_{k-1}+i_{k-2}$

を含み、ｊ＝２である。次にシフトレジスタの内容を適切にフィードバックして内容をα^jで乗算する。これは、（ｍ＋１）ビット内容を左にｊビットだけ循環シフトするのと同じである。（ｋ－２）番目のデータディスクの対応するセクタからのデータもまたレジスタに加算し、すべてのデータディスクの対応するセクタからのデータがレジスタに加算されるまで同様に行なう。次にシフトレジスタの内容を、ｍビット冗長シンボルc_2を生成する変換器１３０に与える。

【００４１】好ましくは、c_3，c_4，…，c_rに対する回路は、上記のように単一クロックサイクルで適切なｊビットシフトを生じさせるフィードバック経路と同様に構成される。

【００４２】（２）　復号化

上記のように、このシステムは、検出されたがデータ符号語におけるＥＣＣシンボルを用いて訂正することができない誤りを訂正する。したがって、誤りのロケーションはわかっている。以下で図８を参照してより詳細に述べるように、現システムを用いて１つまたは複数の誤りを訂正することには、一般的に、重み１、２または３の要素の逆数を発見することが含まれる。

【００４３】重み１のＧＦ（2^m）要素、すなわち１つの非ゼロ係数を有する要素は、１つの１およびｍの０、または１つの０およびｍの１を含む（ｍ＋１）ビット表現に相当する。ｉ番目のビットで１つの１である係数を有する要素α^i＝０００…０１…００について、乗法逆

元α^{-i}は、α^{i}で乗算されたときに$\alpha^{0}=1$を生成する要素である。

【００４４】シンボルα^{0}は（ｍ＋１）ビットシンボルα^{m+1}と同じである。（ｍ＋１）ビット表現を用いると、α^{i}の乗法逆元は、α^{i}のｉ番目の係数を（ｍ＋１）番目の位置に循環シフトする要素である。したがって逆数はα^{m+1-i}である。ｍ＝４の場合、シンボルα^{2}＝００１００は、$\alpha^{5-2}=\alpha^{3}$の乗法逆元を有する。したがって、$\alpha^{2}*\alpha^{3}=\alpha^{5}=\alpha^{0}$である。３ビットを左に循環シフトすることによって$\alpha^{2}$を$\alpha^{3}$で乗算することができる。

【００４５】重み２の要素すなわち２つの１の係数およびｍ－１の０の係数を備える要素に対し、乗法逆元は、

００００...００１１＝$\alpha^{1}+\alpha^{0}$

の逆数から決定される。このシンボルの乗法逆元は、０１０１...１０１０：００００...００１１＊０１０１...１０１０＝０１０１...１０１０＋１０１０...０１００＝１１１１...１１１０または００００...０００１である。したがって逆数は、シンボル$\alpha^{m-1}+\alpha^{m-3}+\alpha^{m-5}+...+\alpha^{3}+\alpha^{1}$であり、このシンボルをδと定めると、

$\delta*(\alpha^{1}+\alpha^{0})=\alpha^{0}$

が得られる。

【００４６】和$\alpha^{1}+\alpha^{0}$を２の累乗に累乗すると以下が生成される。

【００４７】

【数１４】

$$(\alpha^{1}+\alpha^{0})^{2^{0}}=\alpha^{2^{0}}+\alpha^{0}=\alpha^{1}+\alpha^{0}$$

$$(\alpha^{1}+\alpha^{0})^{2^{1}}=\alpha^{2^{1}}+\alpha^{0}=\alpha^{2}+\alpha^{0}$$

$$(\alpha^{1}+\alpha^{0})^{2^{2}}=\alpha^{2^{2}}+\alpha^{0}=\alpha^{4}+\alpha^{0}$$

$$...$$

$$(\alpha^{1}+\alpha^{0})^{2^{m}}=\alpha^{2^{m}}+\alpha^{0}=\alpha^{2^{m}}+\alpha^{0}$$

【００４８】２をＧＦ（ｍ＋１）の原始元とすると、2^{i}ｍｏｄ（ｍ＋１）は１とｍとの間の何らかの値となり得る。したがって、いかなる要素$\alpha^{t}+\alpha^{0}$も、$(\alpha^{1}+\alpha^{0})^{t}$として表わすことがことができ、このｔは０≦ｎ≦ｍの何らかの値について2^{n}である。$(\alpha^{1}+\alpha^{0})^{t}$の逆数は以下のようになる。

【００４９】

【数１５】

$$[(\alpha^{1}+\alpha^{0})^{t}]^{-1}=[\alpha^{1}+\alpha^{0}]^{t*(-1)}=\delta^{t}$$

または

$$(\alpha^{m-1})^{t}+...+(\alpha^{3})^{t}+(\alpha^{1})^{t}$$

【００５０】式中、指数はモジュロｍ＋１である。（ｍ＋１）ビットシンボルを用いると、δを２の累乗に累乗することは、（ｍ＋１）ビットシンボルのビットの置換として行なわれる。ｉ番目のビットの係数すなわち累乗$t=2^{n}$に累乗したδのx^{i}は以下のとおりである。

【００５１】

【数１６】

$$X^{(2^{n}*i)\bmod m+1}$$

【００５２】何らかの重み２の要素$A=\alpha^{h}+\alpha^{v}$、ｈ＜ｖの逆数を決定するために、このシステムは、$\alpha^{h}*\sigma=A$の$\sigma=\alpha^{0}+\alpha^{v-h}$を用いる。したがってσの逆数は以下のようになる。

【００５３】

【数１７】

$$\sigma^{-1}=[\alpha^{0}+\alpha^{v-h}]^{-1}$$

または

$$\delta^{v-h}=\alpha^{1*(v-h)}+\alpha^{3*(v-h)}...+\alpha^{(m-3)*(v-h)}+\alpha^{(m-1)*(v-h)}$$

$$A^{-1}=(\alpha^{h}*\sigma)^{-1}=(\alpha^{h})^{-1}*\sigma^{-1}$$

【００５４】要素α^{h}は重み１を有するため以下のようになる。

【００５５】

【数１８】

$$A^{-1}=(\alpha^{m+1-h})*\delta^{v-h}$$

【００５６】量δ^{v-h}は置換として計算され、積は置換されたビットを（ｍ＋１－ｈ）ビット左に循環シフトさせたものである。

【００５７】ｍ＝４、Ａ＝１０１００または$\alpha^{2}+\alpha^{4}$、ｈ＝２、およびｖ＝４の場合の例は以下のようになる。

【００５８】

【数１９】

$$\sigma=\alpha^0+\alpha^{(4-2)}=\alpha^0+\alpha^2$$

$$\sigma^{-1}=\delta^2$$

または

$$\alpha^{1*2}+\alpha^{3*2}=\alpha^2+\alpha^{6(\mathrm{mod}\,5)}=\alpha^2+\alpha^1$$

$$A^{-1}=(\alpha^{5-2})*(\alpha^2+\alpha^1)=$$

$$\alpha^3*(\alpha^2+\alpha^1)=$$

$$\alpha^0+\alpha^4=10001$$

【００５９】次に挙げるのはこれを検査するための乗法逆元である。

Ａ＊A^{-1}＝１０１００＊１０００１＝

１０１００＋０１０１０＝１１１１０＝０００００１

重み３の要素について、このシステムは０００．．．０００１１１，０００．．．００１０１１，０００．．．０１００１１，（以降同様）の形式の要素の逆数の（ｍ－１）要素ルックアップテーブルを用いる。何らかの重み３の要素Ａの逆数を発見するために（$A=\alpha^a+a^b+\alpha^c$かつａ＞ｂ＞ｃ）、このシステムは最初にＡを、α^{-c}すなわち$\alpha^{(m+1-c)}$で乗算し、以下を形成する。

【００６０】

【数２０】

$$A*\alpha^{-c}=\alpha^{a'}+\alpha^{b'}+1=\gamma$$
$$a'=a-c$$
$$b'=b-c$$

【００６１】これは以下と同じである。

【００６２】

【数２１】

$$\gamma=(\alpha^{a''}+\alpha^1+1)^{b'}$$

$$a''*b'=a' \quad \text{または} \quad a''=\frac{a'}{b'}(\mathrm{mod}\ m+1)$$

【００６３】次に、$\alpha^{a'}+\alpha^1+1$の逆数をルックアップテーブルから検索する。この逆数は要素Ｄであり、γの逆数は$D^{b'}$である。Ａの逆数はしたがって以下のとおりとなる。

【００６４】

【数２２】

$$\alpha^c*D^{b'}=$$
$$\alpha^c*D^{b-c}$$

【００６５】式中D^{b-c}は置換として計算され、α^cによる乗算は循環シフトである。

（３）　システム

次に図８を参照して、符号器６１０は、ｋのデータディスク６１２からの対応するセクタを符号化して、ｒの冗長ディスク６１４に記録するためにｒの冗長セクタを生

成する。この符号器は図４－７を参照して先に述べた回路100_rを用いて、冗長シンボルc_0，c_1，．．．，c_{r-1}を生成する。

【００６６】復号器６２０および誤り訂正プロセッサ６３０は、ｋのデータディスクにストアされたデータにおける誤りを訂正する。この誤りは、データディスクにストアされたＥＣＣシンボルを用い従来のやり方で検出される。

【００６７】復号器６２０は、誤りのないｍビットセクタの（ｍ＋１）ビット表現を用い、誤りのあるセクタについてはｍ＋１のゼロを用いてカラムを復号化する。したがって誤りのあるセクタはカラム復号化については読出されない。ｑの誤りのあるセクタがあるならば（ｑは１つ以上の冗長セクタを含み得る）、このシステムはｑの１次方程式の組を解いて誤りのある情報を回復する。

【００６８】この復号器は回路100_rを用いて冗長シンボルC_rを以下のようにしてc'_rとして再生する。

【００６９】

【数２３】

$$c'_r=i'_0+(\alpha^{r*1})i'_1+\ldots+\alpha^{r*k-1}i'_{k-1}$$

【００７０】式中、何らかのｉ＝０、１、…、ｋ－１に対するi'_iは、ｉ番目のデータディスクから読出した誤りのないデータであり、もしデータが誤りを含んでいればすべて０である。復号器は、シンドローム発生器６２１において従来の態様で、誤りのない冗長ディスクに対応するシンドロームS_j、ｊ＝０、１…ｒ－１を生成する。

【００７１】１つの誤りを訂正するためのシンドロームは、

$S_j=(\alpha^j)^p*i'_p$

であり、i'_pは、ｐ番目のディスクと関連する誤りパターンである。誤りの値は、

$i'_p=S_j*(\alpha^{jp})^{-1}$

を解く、誤り値プロセッサ６２２によって回復される。α^jの重みは１であるので、上記のように、α^{jp}の重みは１であり（$(\alpha^{jp})^{-1}$は（α^{m+1-jp}）である。プロセッサはこのようにして容易に、重み１のサブプロセッサ６２４を用いて乗法逆元を決定し、逆数を循環シフトとしてS_jで乗算する。

【００７２】二重の誤りを訂正するためのシンドロームは以下のとおりである。

【００７３】

【数２４】

$$S_j=(\alpha^j)^p*i'_p+(\alpha^j)^q i'_q$$
$$S_{j+1}=(\alpha^{j+1})^p*i'_p+(\alpha^{j+1})^q i'_q$$

【００７４】式中i'_qはｑ番目のディスクと関連する誤りパターンである。誤りの値はしたがって以下のよう

にして決定される。

【００７５】

【数２５】

$$i'_p=\frac{\left(S_j*(\alpha^{j+1})^q\right)+\left(S_{j+1}*(\alpha^j)^q\right)}{\left[(\alpha^j)^p*(\alpha^{j+1})^q\right]+\left[(\alpha^{j+1})^p*(\alpha^{j+1})^q\right]}$$

$$i'_q=\frac{\left(S_j*(\alpha^{j+1})^p\right)+\left(S_{j+1}*(\alpha^j)^q\right)}{\left[(\alpha^j)^p*(\alpha^{j+1})^q\right]+\left[(\alpha^{j+1})^p*(\alpha^{j+1})^q\right]}$$

【００７６】分母は、αの累乗の２つの積の和、すなわち$\alpha^a+\alpha^b$である。この和の重みは、もしａ＝ｂであれば０であり、もしａ≠ｂであれば２である。重み２の分母については、$\alpha^a+\alpha^b$の逆数は上記のように、重み２のサブプロセッサ６２５を用いて容易に決定することができる。

【００７７】システムは、３つ以上の誤りを訂正するために、連続冗長ディスクに対応するシンドロームを用いる。式の組を解くために、システムは以下の行列式の逆数を決定する。

【００７８】

【数２６】

$$D=\begin{vmatrix}\alpha^{j*e_0} & \alpha^{j*e_1} & \cdots & \alpha^{j*e_z}\\ \alpha^{(j+1)*e_0} & \alpha^{(j+1)*e_1} & \cdots & \alpha^{(j+1)*e_z}\\ \vdots & & & \\ \alpha^{(j+z)*e_0} & \alpha^{(j+z)*e_1} & \cdots & \alpha^{(j+z)*e_z}\end{vmatrix}$$

または

$$D=J*\begin{vmatrix}1 & 1 & \cdots & 1\\ \alpha^{e_0} & \alpha^{e_1} & \cdots & \alpha^{e_z}\\ \vdots & & & \\ \alpha^{z*e_0} & \alpha^{z*e_1} & \cdots & \alpha^{z*e_z}\end{vmatrix}=JV$$

式中

$$J=[(\alpha^{j*e_0})*(\alpha^{j*e_1})*\ldots*(\alpha^{j*e_z})$$

であり、Ｖは既知のヴァンデルモンドの行列式。

【００７９】ヴァンデルモンドの行列式の値は、すべての$e_0\leqq a\leqq e_{z-1}$および$e_1\leqq b\leqq e_z$について以下の項の積に等しい。

【００８０】

【数２７】

【００８１】したがって、行列式Ｄの値は、因子の逆数の積である以下の逆数である。

【００８２】

【数２８】

$$\frac{(\alpha^{e_a}+\alpha^{e_b})}{J*(\alpha^{e_0}+\alpha^{e_1})*(\alpha^{e_0}+\alpha^{e_2})*\ldots*(\alpha^{e_{z-1}}+\alpha^{e_z})}$$

【００８３】因子Ｊは、重み１のアルファの累乗の連続の積であり、Ｊはしたがって、重み１のアルファの累乗である。Ｊの逆数は、α^{m+1-j}として容易に決定することができる。ヴァンデルモンドの行列式の因子は各々重み２であり、その逆数は上記のように重み２のサブプロセッサ６２５を用いて容易に決定される。

【００８４】行列式ＪＶは、Ｄｅｔｊ／ＪＶによって決定される誤りの値を解くために用いられる分母である。このＤｅｔ$_j$は、ｊ番目のカラムを適切なシンドロームで置き換えた行列式Ｄである。したがってＤｅｔ$_j$は項

$(\alpha^a+\alpha^b)*S_j$

の和に等しく、各積は、S_jを循環シフトさせたもの２つの和として計算される。

【００８５】ｚの連続するシンドロームを生成するのに十分な数の連続した誤りのない冗長ディスクがなければ、分母における行列式はヴァンデルモンドの行列式ではない。生成される因子はしたがって、２よりも大きな重みを有する。重み３の因子については、重み３のサブプロセッサ６２６が上記のように逆数を決定する。３よりも大きな重みを有する項については、重み＞３のサブプロセッサ６２７がユークリッドの互除法を用いて反復的に$a(x)*b(x)+q(x)*(x^m+x^{m+1}+\ldots 1+x+1)$を解く。ｂ（ｘ）は逆数である。

【００８６】代わりに、このシステムは、カラムＥＣＣ

について非原始のリードソロモン符号を用いてもよく、カラム符号器および復号器はしたがって、本明細書に引用により援用する、FettweisおよびHassnerによる、「ハードウェアの複雑性が低い複合リードソロモン符号器およびシンドローム発生器（A Combined Reed-Solomon Encoder and Syndrome Generator With Small Hardware Complexity）」、（Proceeding of the 1992 IEEE International Symposium On Circuits and Systems, San Diego, California, May 10-13 1993, pp. 1871-1874）において述べられているように、ハードウェアを共有することができる。たとえば、符号器および復号器は、乗算器、シフトレジスタおよび加算器を共有してもよい。非原始のリードソロモン符号を用いるならば、最大で全ディスクはｍであり、すなわちｋ＋ｒ≦ｍである。

【００８７】非原始のリードソロモン符号は、ｔ＋１の連続する根を有する生成多項式

$g(x) = (x+\alpha^0)(x+\alpha^1)\ldots(x+\alpha^t)$

を有する。非原始のリードソロモン符号については、シンドロームは、単にデータディスクではなくディスクすべてからの検索されたシンボルに基づき決定される。したがってこのシンドロームは、以下のようになる。

【００８８】

【数２９】

$$S_j = i'_0 + (\alpha^{j*1})i'_1 + \ldots + (\alpha^{j*k+r-2})i'_{k+r-2}(\alpha^{j*k+r-1})i_{k+r-1}$$

$$j = 0, 1, \ldots, t$$

【００８９】いかなる数の誤り $z \leq r$ についても、常に r の連続するシンドロームが利用できる。実際に、最初の z のシンドロームを使用することができる。誤りの値を解くために逆数が必要とされる行列式はしたがってヴァンデルモンドの行列式である。したがって上記のように、行列式の逆数は容易に、各々が重み２を有する、因子 $\alpha^a+\alpha^b$ の逆数をともに乗算することによって発見される。誤りの値を決定するための計算は本質的には、いかなる数の誤りについても同じである。こうして、システムは容易に r の誤りまで、重みが２よりも大きな何らかの要素の逆数を発見せずとも、訂正することができる。

【００９０】上記のシステムは、ｍビットシンボルのカラムを（ｍ＋１）シンボルとして符号化し、ｍはディスクセクタまたはテープブロックにおけるビット数と同じまたはこれよりも大きなものとして選択することができる。カラム符号化についての既知の先行技術のシステムでは、このような大きなｍビットシンボルの使用は、必要なガロア体乗算器、および $GF(2^m)$ における乗法逆元を発見するための回路が複雑なために本質的に禁じられている。このシステムはこうした演算を簡単にしており、したがって本質的に任意の大きな値のｍを使用することができる。上記のように、ｍの値は、必要に応じてより大きなｍの値を含むように拡張できる、図１から３の表から選択される。

【００９１】上記の説明は、本発明の特定的な実施例に限定されている。しかしながら、循環シフト演算を行なうシフトレジスタの種々の構成の使用、ソフトウェア、ハードウェアまたはファームウェアにおいて種々の演算を行なうこと、または種々の単一目的のプロセッサを１つ以上の多重目的プロセッサに組合せることなどの変形および修正を本発明に施し、その利点のいくつかまたはすべてを得ることができることが明らかであろう。したがって、前掲の請求項の目的は、本発明の精神および範囲内のこうしたすべての変形および修正を網羅することである。

【図面の簡単な説明】

【図１】ｍの選択された値の表の図である。

【図２】図１に続く表の図である。

【図３】図２に続く表の図である。

【図４】冗長シンボル c_r を生成するための回路の機能ブロック図である。

【図５】冗長シンボル c_0 生成するための回路の機能ブロック図である。

【図６】冗長シンボル c_1 を生成するための回路の機能ブロック図である。

【図７】冗長シンボル c_2 を生成するための好ましい回路の機能ブロック図である。

【図８】図４から７の回路を組込んだ符号化／復号化システムの機能ブロック図である。

【符号の説明】

６１０　符号器

６２０　復号器

６３０　誤り訂正プロセッサ

【図１】

既約多項式 $x^m + x^{m-1} + x^{m-2} + \dots + x^3 + x^2 + x + 1$ についての
GF(2^m) に対するmの値

2	4	10	12	18	28	36	52	58	60
66	82	100	106	130	138	148	162	172	178
180	196	210	226	268	292	316	346	348	372
378	388	418	420	442	460	466	490	508	522
540	546	556	562	586	612	618	652	658	660
676	700	708	756	772	786	796	820	826	828
852	858	876	882	906	940	946	1018	1060	1090
1108	1116	1122	1170	1186	1212	1228	1236	1258	1276
1282	1290	1300	1306	1372	1380	1426	1450	1452	1482
1492	1498	1522	1530	1548	1570	1618	1620	1636	1666
1668	1692	1732	1740	1746	1786	1860	1866	1876	1900
1906	1930	1948	1972	1978	1986	1996	2026	2028	2052
2068	2082	2098	2130	2140	2212	2220	2236	2242	2266
2268	2292	2308	2332	2338	2356	2370	2388	2436	2458
2466	2476	2530	2538	2548	2556	2578	2620	2658	2676
2682	2692	2698	2706	2740	2788	2796	2802	2818	2836
2842	2850	2860	2908	2938	2956	2962	3010	3018	3036
3066	3082	3186	3202	3252	3298	3306	3322	3346	3370
3412	3460	3466	3468	3490	3498	3516	3532	3538	3546
3556	3570	3580	3612	3636	3642	3658	3676	3690	3700
3708	3732	3778	3796	3802	3850	3852	3876	3906	3916
3922	3930	3946	3988	4002	4012	4018	4020	4090	4092
4098	4132	4138	4156	4218	4228	4242	4252	4258	4260
4282	4348	4356	4362	4372	4396	4450	4482	4492	4506
4516	4546	4602	4620	4636	4690	4722	4786	4788	4812
4876	4932	4956	4972	4986	5002	5010	5050	5058	5076
5098	5106	5146	5170	5178	5188	5226	5260	5308	5332
5386	5442	5476	5482	5500	5506	5556	5562	5572	5650
5658	5682	5692	5700	5716	5740	5748	5778	5812	5826
5842	5850	5868	5922	5938	5986	6010	6028	6052	6066
6100	6130	6172	6196	6202	6210	6228	6268	6276	6298
6316	6322	6372	6378	6388	6396	6468	6490	6546	6618
6636	6652	6658	6690	6700	6708	6732	6762	6778	6780
6802	6826	6828	6868	6882	6898	6906	6916	6946	6948
6970	7012	7018	7026	7042	7068	7108	7186	7210	7218
7228	7236	7242	7252	7282	7306	7330	7348	7410	7450
7458	7476	7498	7506	7516	7522	7540	7546	7548	7572
7588	7602	7620	7642	7668	7690	7716	7756	7788	7828
7852	7876	7882	7900	7906	7932	7948	8052	8068	8092
8116	8122	8146	8170	8178	8218	8220	8236	8242	8268
8290	8292	8362	8386	8428	8442	8466	8538	8562	8572
8596	8626	8668	8676	8692	8698	8730	8740	8746	8802

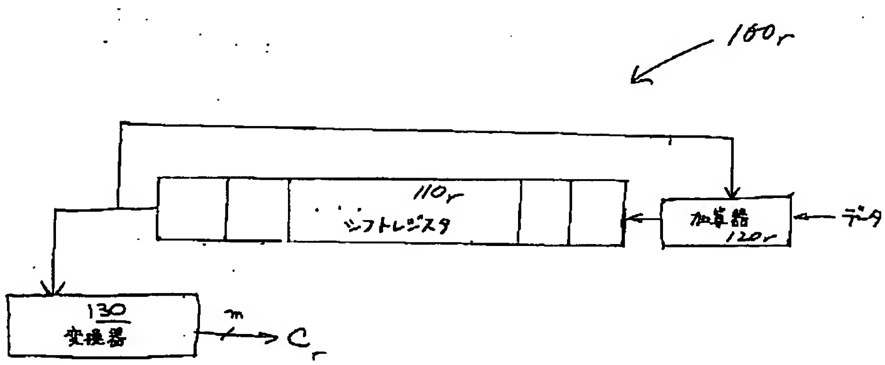
（図2に続く）

【図２】

8818	8820	8836	8860	8866	8922	8932	8962	8970	9010
9028	9058	9172	9180	9202	9220	9226	9282	9292	9322
9340	9348	9370	9396	9418	9420	9436	9466	9490	9532
9538	9546	9586	9612	9618	9628	9642	9660	9676	9732
9748	9802	9850	9858	9882	9900	9906	9922	9940	9948
10036	10066	10068	10090	10092	10098	10132	10138	10140	10162
10180	10252	10258	10266	10300	10330	10356	10426	10458	10476
10498	10500	10588	10612	10666	10690	10708	10722	10732	10788
10836	10852	10858	10860	10866	10882	10890	10908	10948	10972
10978	10986	11002	11026	11068	11082	11092	11130	11170	11196
11212	11260	11316	11436	11442	11482	11548	11578	11586	11620
11676	11698	11716	11778	11788	11812	11820	11826	11866	11908
11932	11938	11980	11986	12010	12042	12106	12148	12156	12196
12202	12210	12226	12250	12252	12268	12276	12300	12322	12346
12372	12378	12412	12436	12490	12538	12546	12588	12610	12612
12618	12636	12652	12658	12738	12756	12762	12780	12820	12828
12898	12906	12916	12922	12940	12978	13036	13042	13108	13146
13162	13186	13228	13290	13330	13338	13396	13410	13450	13468
13476	13522	13612	13618	13626	13690	13708	13722	13756	13762
13828	13858	13876	13882	13900	13906	13930	13932	13996	14010
14050	14106	14172	14220	14242	14340	14386	14388	14410	14418
14460	14532	14548	14556	14620	14626	14628	14652	14668	14698
14716	14722	14740	14746	14770	14796	14812	14820	14826	14842
14850	14866	14868	14890	14922	14938	14946	14956	15012	15052
15060	15076	15082	15090	15100	15106	15130	15138	15148	15172
15186	15226	15258	15268	15298	15330	15348	15372	15412	15426
15442	15460	15580	15628	15660	15666	15682	15730	15738	15748
15772	15786	15796	15802	15858	15906	15922	15970	16066	16068
16138	16186	16188	16228	16252	16300	16332	16338	16348	16362
16380	16420	16426	16452	16476	16492	16546	16572	16602	16618
16650	16692	16746	16762	16786	16828	16842	16882	16900	16930
16962	16978	16980	17010	17020	17026	17052	17076	17092	17098
17122	17188	17202	17298	17316	17332	17386	17388	17442	17476
17482	17508	17572	17578	17596	17626	17668	17746	17748	17788
17826	17836	17850	17890	17908	17922	17938	17956	17980	17988
18012	18042	18058	18076	18148	18180	18228	18250	18268	18412
18426	18442	18460	18492	18586	18636	18700	18730	18748	18756
18772	18786	18858	18868	18898	18916	18946	18972	18978	19012
19036	19050	19068	19138	19140	19156	19162	19180	19218	19236
19258	19300	19332	19372	19378	19386	19402	19426	19468	19482
19500	19506	19540	19570	19596	19602	19660	19708	19716	19738
19762	19812	19852	19866	19890	19948	19962	19972	19978	19996
20028	20050	20106	20116	20122	20148	20172	20218	20260	20268
20322	20340	20356	20442	20476	20506	20508	20532	20548	20610
20626	20692	20716	20730	20748	20770	20772	20788	20898	20938
20962	20980	21010	21012	21018	21058	21066	21100	21106	21138

（図3に続く）

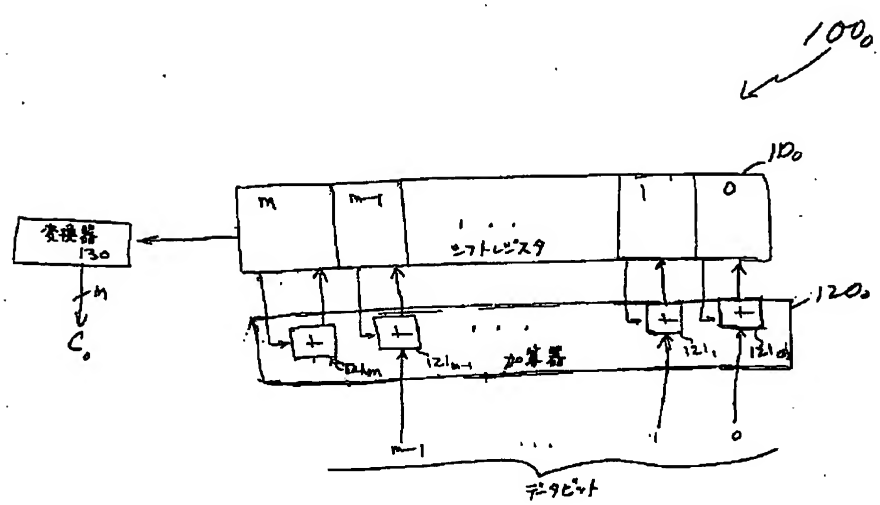
【図４】



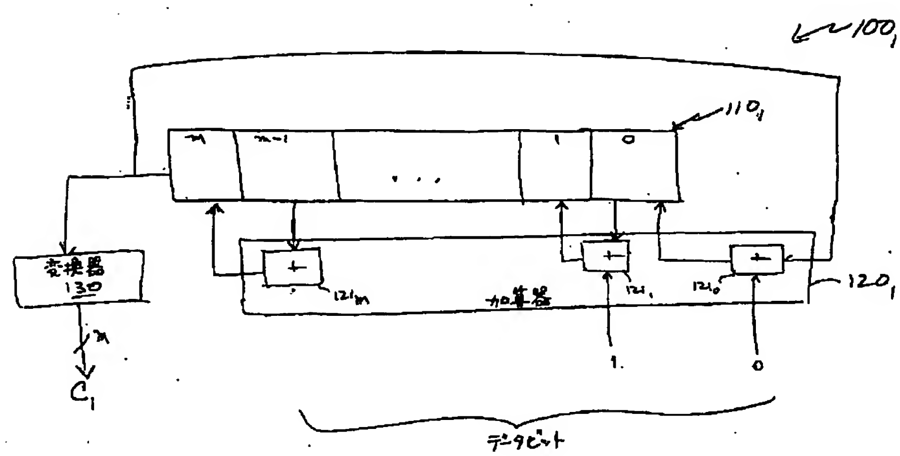
【図３】

21156	21162	21178	21186	21210	21220	21226	21268	21316	21340
21346	21378	21396	21418	21466	21490	21492	21522	21556	21562
21586	21588	21610	21612	21660	21682	21700	21738	21772	21802
21820	21858	21892	22002	22012	22036	22066	22090	22108	22122
22132	22156	22170	22188	22228	22258	22276	22282	22306	22348
22396	22468	22482	22500	22530	22548	22612	22618	22620	22636
22642	22668	22690	22708	22716	22738	22786	22852	22858	22860
22876	22900	22972	23002	23020	23026	23028	23052	23098	23116
23188	23196	23250	23332	23338	23356	23370	23458	23508	23530
23538	23548	23562	23602	23626	23628	23668	23740	23746	23788
23812	23818	23826	23868	23898	23908	23916	23956	24018	24028
24042	24076	24082	24106	24108	24178	24196	24202	24228	24316
24370	24378	24412	24418	24442	24498	24508	24532	24546	24658
24676	24682	24690	24708	24732	24748	24762	24780	24820	24850
24858	24906	24916	24922	24970	24978	24988	25012	25036	25146
25162	25188	25218	25228	25236	25242	25252	25306	25348	25356
25372	25452	25468	25522	25540	25578	25588	25602	25642	25666
25692	25716	25732	25746	25770	25866	25930	25932	25996	26002
26020	26052	26098	26106	26140	26170	26188	26236	26250	26260
26266	26308	26338	26356	26386	26458	26500	26538	26556	26572
26596	26626	26668	26682	26692	26698	26716	26722	26812	26820
26860	26946	26986	27010	27042	27058	27060	27066	27076	27090
27106	27178	27196	27252	27258	27298	27396	27436	27508	27538
27580	27610	27652	27700	27732	27772	27778	27802	27826	27850
27892	27900	27916	27940	27946	28018	28026	28050	28098	28108
28122	28162	28180	28210	28228	28276	28282	28306	28308	28348
28386	28410	28428	28476	28492	28498	28516	28540	28546	28548
28570	28572	28578	28596	28602	28618	28642	28660	28722	28770
28812	28836	28842	28858	28908	28932	28948	28978	29020	29026
29058	29076	29100	29122	29130	29146	29172	29220	29242	29250
29332	29338	29362	29386	29388	29410	29428	29436	29442	29452
29482	29500	29530	29572	29586	29668	29716	29722	29740	29788
29836	29850	29866	29916	29988	30010	30012	30028	30058	30108
30138	30180	30196	30202	30210	30252	30292	30306	30322	30340
30346	30388	30466	30468	30490	30508	30516	30538	30556	30636
30642	30660	30676	30706	30762	30780	30802	30828	30850	30852
30858	30868	30892	30930	30940	30970	31012	31018	31068	31090
31138	31146	31180	31252	31258	31266	31276	31306	31356	31378
31386	31396	31468	31516	31530	31546	31642	31666	31698	31858
31882	31890	31906	31956	31962	31972	32002	32026	32028	32058
32068	32076	32082	32098	32116	32140	32188	32202	32212	32236
32260	32308	32322	32340	32362	32370	32380	32410	32442	32466
32490	32506	32530	32532	32562	32572	32586	32602	32652	32692
32716	32748	32770	32788	32796	32842	32908	32916	32932	32938
32940	32986								

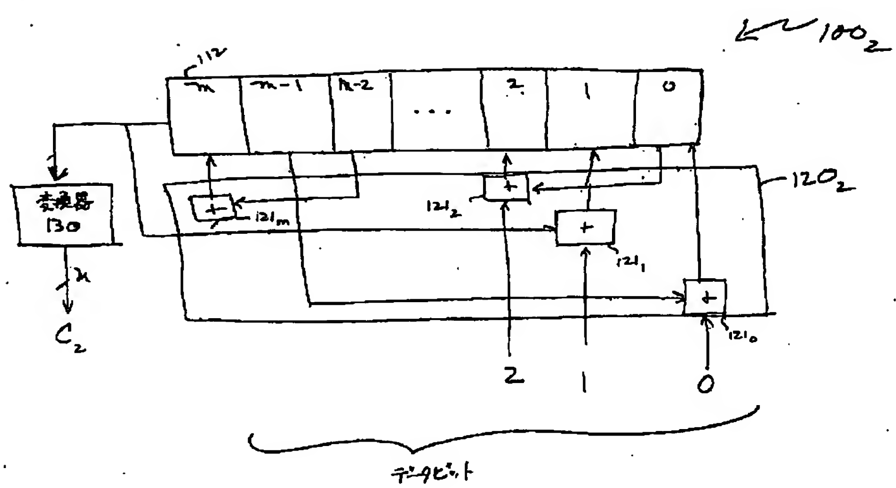
【図５】



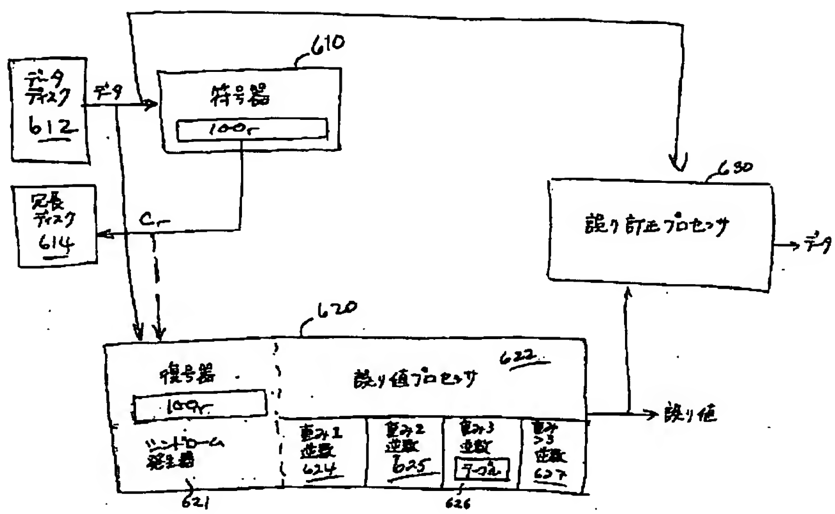
【図６】


100₁
110₁
m
m-1
1
0
変換器
130
121m
加算器
121₁
121₀
120₁
n
C₁
1
0
データビット

【図７】


100₂
112
m
m-1
m-2
2
1
0
変換器
130
121m
121₂
121₁
120₂
121₀
n
C₂
2
1
0
データビット

【図８】


610
データディスク 612
データ
符号器
100r
冗長ディスク 614
Cr
630
誤り訂正プロセッサ
データ
620
復号器
100r
シンドローム発生器
621
誤り値プロセッサ
622
重み1逆数 624
重み2逆数 625
重み3逆数
テーブル
626
重み>3逆数 627
誤り値